Publications of Martin Krejca

This document lists all peer-reviewed publications of Martin Krejca, Chair for Algorithm Engineering, Hasso Plattner Institute, Potsdam, Germany. This listing was automatically generated on October 15, 2020. An up-to-date version is available online at hpi.de/friedrich/docs/publist/krejca.pdf.

Journal articles

[1] Doerr, B., Krejca, M. S., Significance-based Estimation-of-Distribution Algorithms. In: *Transactions on Evolutionary Computation*, 2020.

Estimation-of-distribution algorithms (EDAs) are randomized search heuristics that create a probabilistic model of the solution space, which is updated iteratively, based on the quality of the solutions sampled according to the model. As previous works show, this iteration-based perspective can lead to erratic updates of the model, in particular, to bit-frequencies approaching a random boundary value. In order to overcome this problem, we propose a new EDA based on the classic compact genetic algorithm (cGA) that takes into account a longer history of samples and updates its model only with respect to information which it classifies as statistically significant. We prove that this significance-based compact genetic algorithm (sig-cGA) optimizes the commonly regarded benchmark functions OneMax, LeadingOnes, and BinVal all in quasilinear time, a result shown for no other EDA or evolutionary algorithm so far. For the recently proposed scGA – an EDA that tries to prevent erratic model updates by imposing a bias to the uniformly distributed model – we prove that it optimizes OneMax only in a time exponential in its hypothetical population size. Similarly, we show that the convex search algorithm cannot optimize OneMax in polynomial time.

[2] Krejca, M. S., Witt, C., Lower Bounds on the Run Time of the Univariate Marginal Distribution Algorithm on OneMax. In: *Theoretical Computer Science* 832, pp. 143–165, 2020.

The Univariate Marginal Distribution Algorithm (UMDA) – a popular estimation-of-distribution algorithm – is studied from a run time perspective. On the classical OneMax benchmark function on bit strings of length n, a lower bound of $\Omega(\lambda + \mu\sqrt{n} + n \log n)$, where μ and λ are algorithm-specific parameters, on its expected run time is proved. This is the first direct lower bound on the run time of UMDA. It is stronger than the bounds that follow from general black-box complexity theory and is matched by the run time of many evolutionary algorithms. The results are obtained through advanced analyses of the stochastic change of the frequencies of bit values maintained by the algorithm, including carefully designed potential functions. These techniques may prove useful in advancing the field of run time analysis for estimation-of-distribution algorithms in general.

[3] Kötzing, T., Krejca, M. S., First-hitting times under drift. In: *Theoretical Computer Science* 796, pp. 51–69, 2019.

For the last ten years, almost every theoretical result concerning the expected run time of a randomized search heuristic used drift theory, making it the arguably most important tool in this domain. Its success is due to its ease of use and its powerful result: drift theory allows the user to derive bounds on the expected first-hitting time of a random process by bounding expected local changes of the process – the drift. This is usually far easier than bounding the expected first-hitting time directly. Due to the widespread use of drift theory, it is of utmost importance to have the best drift theorems possible. We improve the fundamental additive, multiplicative, and variable drift theorems by stating them in a form as general as possible and providing examples of why the restrictions we keep are still necessary. Our additive drift theorem for upper bounds only requires the process to be lower-bounded, that is, we remove unnecessary restrictions like a finite, discrete, or bounded state space. As corollaries, the same is true for our upper bounds in the case of variable and multiplicative drift. By bounding the step size of the process, we derive new lower-bounding multiplicative and variable drift theorems. Last, we also state theorems that are applicable when the process has a drift of 0, by using a drift on the variance of the process.

[4] Trubenova, B., Kötzing, T., Krejca, M. S., Lehre, P. K., Surfing on the seascape: Adaptation in a changing environment. In: *Evolution: International Journal of Organic Evolution* 73, pp. 1356–1374, 2019.

The environment changes constantly at various time scales and, in order to survive, species need to keep adapting. Whether these species succeed in avoiding extinction is a major evolutionary question. Using a multilocus evolutionary model of a mutation-limited population adapting under strong selection, we investigate the effects of the frequency of environmental fluctuations on adaptation. Our results rely on an "adaptive-walk" approximation and use mathematical methods from evolutionary computation theory to investigate the interplay between fluctuation frequency, the similarity of environments, and the number of loci contributing to adaptation. First, we assume a linear additive fitness function, but later generalize our results to include several types of epistasis. We show that frequent environmental changes prevent populations from reaching a fitness peak, but they may also prevent the large fitness loss that occurs after a single environmental change. Thus, the population can survive, although not thrive, in a wide range of conditions. Furthermore, we show that in a frequently changing environment, the similarity of threats that a population faces affects the level of adaptation that it is able to achieve. We check and supplement our analytical results with simulations.

[5] Friedrich, T., Krejca, M. S., Rothenberger, R., Arndt, T., Hafner, D., Kellermeier, T., Krogmann, S., Razmjou, A., Routing for On-Street Parking Search using Probabilistic Data. In: *AI Communications* 32, pp. 113–124, 2019.

A significant percentage of urban traffic is caused by the search for parking spots. One possible approach to improve this situation is to guide drivers along routes which are likely to have free parking spots. The task of finding such a route can be modeled as a probabilistic graph problem which is NP-complete. Thus, we propose heuristic approaches for solving this problem and evaluate them experimentally. For this, we use probabilities of finding a parking spot, which are based on publicly available empirical data from TomTom International B.V. Additionally, we propose a heuristic that relies exclusively on conventional road attributes. Our experiments show that this algorithm comes close to the baseline by a factor of 1.3 in our cost measure. Last, we complement our experiments with results from a field study, comparing the success rates of our algorithms against real human drivers.

[6] Friedrich, T., Kötzing, T., Krejca, M. S., Unbiasedness of Estimation-of-Distribution Algorithms. In: *Theoretical Computer Science* 785, pp. 46–59, 2019.

In the context of black-box optimization, black-box complexity is used for understanding the inherent difficulty of a given optimization problem. Central to our understanding of nature-inspired search heuristics in this context is the notion of unbiasedness. Specialized black-box complexities have been developed in order to better understand the limitations of these heuristics – especially of (population-based) evolutionary algorithms (EAs). In contrast to this, we focus on a model for algorithms explicitly maintaining a probability distribution over the search space: so-called estimation-of-distribution algorithms (EDAs). We consider the recently introduced n-Bernoulli-λ-EDA framework, which subsumes, for example, the commonly known EDAs PBIL, UMDA, λ-MMAS$_{\mathrm{IB}}$, and cGA. We show that an n-Bernoulli-λ-EDA is unbiased if and only if its probability distribution satisfies a certain invariance property under isometric automorphisms of $[0,1]^n$. By restricting how an n-Bernoulli-λ-EDA can perform an update, in a way common to many examples, we derive conciser characterizations, which are easy to verify. We demonstrate this by showing that our examples above are all unbiased.

[7] Dang, D.-C., Friedrich, T., Kötzing, T., Krejca, M. S., Lehre, P. K., Oliveto, P. S., Sudholt, D., Sutton, A. M., Escaping Local Optima Using Crossover with Emergent Diversity. In: *Transactions on Evolutionary Computation* 22, pp. 484–497, 2018.

Population diversity is essential for the effective use of any crossover operator. We compare seven commonly used diversity mechanisms and prove rigorous run time bounds for the $(\mu+1)$ GA using uniform crossover on the fitness function $Jump_k$. All previous results in this context only hold for unrealistically low crossover probability $p_c = O(k/n)$, while we give analyses for the setting of constant $p_c < 1$ in all but one case. Our bounds show a dependence on the problem size n, the jump length k, the population size μ, and the crossover probability p_c. For the typical case of constant $k > 2$ and constant p_c, we can compare the resulting expected optimisation times for different diversity mechanisms assuming an optimal choice of μ: $O(n^{k-1})$ for duplicate elimination/minimisation, $O(n^2 \log n)$ for maximising the convex hull, $O(n \log n)$ for det. crowding (assuming $p_c = k/n$), $O(n \log n)$ for maximising the Hamming distance, $O(n \log n)$ for fitness sharing, $O(n \log n)$ for the single-receiver island model. This proves a sizeable advantage of all variants of the $(\mu+1)$ GA compared to the (1+1) EA, which requires $\Theta(n^k)$. In a short empirical study we confirm that the asymptotic differences can also be observed experimentally.

[8] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., The Compact Genetic Algorithm is Efficient under Extreme Gaussian Noise. In: *Transactions on Evolutionary Computation* 21, pp. 477–490, 2017.

Practical optimization problems frequently include uncertainty about the quality measure, for example due to noisy evaluations. Thus, they do not allow for a straightforward application of traditional optimization techniques. In these settings, randomized search heuristics such as evolutionary algorithms are a popular choice because they are often assumed to exhibit some kind of resistance to noise. Empirical evidence suggests that some algorithms, such as estimation of distribution algorithms (EDAs) are robust against a scaling of the noise intensity, even without resorting to explicit noise-handling techniques such as resampling. In this paper, we want to support such claims with mathematical rigor. We introduce the concept of graceful scaling in which the run time of an algorithm scales polynomially with noise intensity. We study a monotone fitness function over binary strings with additive noise taken from a Gaussian distribution. We show that myopic heuristics cannot efficiently optimize the function under arbitrarily intense noise without any explicit noise-handling. Furthermore, we prove that using a population does not help. Finally we show that a simple EDA called the compact Genetic Algorithm can overcome the shortsightedness of mutation-only heuristics to scale gracefully with noise. We conjecture that recombinative genetic algorithms also have this property.

[9] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., Robustness of Ant Colony Optimization to Noise. In: *Evolutionary Computation* 24, pp. 237–254, 2016.

Recently Ant Colony Optimization (ACO) algorithms have been proven to be efficient in uncertain environments, such as noisy or dynamically changing fitness functions. Most of these analyses focus on combinatorial problems, such as path finding. We analyze an ACO algorithm in a setting where we try to optimize the simple OneMax test function, but with additive posterior noise sampled from a Gaussian distribution. Without noise the classical $(\mu+1)$-EA outperforms any ACO algorithm, with smaller μ being better; however, with large noise, the $(\mu+1)$-EA fails, even for high values of μ (which are known to help against small noise). In this paper we show that ACO is able to deal with arbitrarily large noise in a graceful manner, that is, as long as the evaporation factor p is small enough dependent on the parameter δ^2 of the noise and the dimension n of the search space ($p = o(1/(n(n+\delta \log n)^2 \log n))$), optimization will be successful.

Conference papers

[10] Doerr, B., Krejca, M. S., The Univariate Marginal Distribution Algorithm Copes Well With Deception and Epistasis. In: *Evolutionary Computation in Combinatorial Optimisation (EvoCOP)*, pp. 51–66, 2020. Best-Paper Award.

In their recent work, Lehre and Nguyen (FOGA 2019) show that the univariate marginal distribution algorithm (UMDA) needs time exponential in the parent populations size to optimize the DeceivingLeadingBlocks (DLB) problem. They conclude from this result that univariate EDAs have difficulties with deception and epistasis. In this work, we show that this negative finding is caused by an unfortunate choice of the parameters of the UMDA. When the population sizes are chosen large enough to prevent genetic drift, then the UMDA optimizes the DLB problem with high probability with at most $\lambda(\frac{n}{2} + 2e \ln n)$ fitness evaluations. Since an offspring population size λ of order $n \log n$ can prevent genetic drift, the UMDA can solve the DLB problem with $O(n^2 \log n)$ fitness evaluations. In contrast, for classic evolutionary algorithms no better run time guarantee than $O(n^3)$ is known, so our result rather suggests that the UMDA can cope well with deception and epistatis. Together with the result of Lehre and Nguyen, our result for the first time rigorously proves that running EDAs in the regime with genetic drift can lead to drastic performance losses.

[11] Doerr, B., Krejca, M. S., Bivariate Estimation-of-Distribution Algorithms Can Find an Exponential Number of Optima. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 796–804, 2020.

Finding a large set of optima in a multimodal optimization landscape is a challenging task. Classical population-based evolutionary algorithms (EAs) typically converge only to a single solution. While this can be counteracted by applying niching strategies, the number of optima is nonetheless trivially bounded by the population size. Estimation-of-distribution algorithms (EDAs) provide an alternative approach by maintaining a probabilistic model of the solution space instead of an explicit population. Such a model has the benefit of being able to implicitly represent a solution set that is far larger than any realistic population size. To support the study of how optimization algorithms handle large sets of optima, we propose the test function EqualBlocksOneMax (EBOM). It has an easy to optimize fitness landscape, however, with an exponential number of optima. We show that the bivariate EDA mutual-information-maximizing input clustering (MIMIC), without any problem-specific modification, quickly generates a model that behaves very similarly to a theoretically ideal model for that function, which samples each of the exponentially many optima with the same maximal probability.

[12] Friedrich, T., Krejca, M. S., Lagodzinski, J. A. G., Rizzo, M., Zahn, A., Memetic Genetic Algorithms for Time Series Compression by Piecewise Linear Approximation. In: *International Conference on Neural Information Processing (ICONIP)*, 2020.

Time series are sequences of data indexed by time. Such data are collected in various domains, often in massive amounts, such that storing them proves challenging. Thus, time series are commonly stored in a compressed format. An important compression approach is piecewise linear approximation (PLA), which only keeps a small set of time points and interpolates the remainder linearly. Picking a subset of time points such that the PLA minimizes the mean squared error to the original time series is a challenging task, naturally lending itself to heuristics. We propose the piecewise linear approximation genetic algorithm (PLA-GA) for compressing time series by PLA. The PLA-GA is a memetic $(\mu+\lambda)$ GA that makes use of two distinct operators tailored to time series compression. First, we add special individuals to the initial population that are derived using established PLA heuristics. Second, we propose a novel local search operator that greedily improves a compressed time series. We compare the PLA-GA empirically with existing evolutionary approaches and with a deterministic PLA algorithm, known as Bellman's algorithm, that is optimal for the restricted setting of sampling. In both cases, the PLA-GA approximates the original time series better and quicker. Further, it drastically outperforms Bellman's algorithm with increasing instance size with respect to run time until finding a solution of equal or better quality – we observe speed-up factors between 7 and 100 for instances of 90,000 to 100,000 data points.

[13] Peters, J., Stephan, D., Amon, I., Gawendowicz, H., Lischeid, J., Salabarria, J., Umland, J., Werner, F., Krejca, M. S., Rothenberger, R., Kötzing, T., Friedrich, T., Mixed Integer Programming versus Evolutionary Computation for Optimizing a Hard Real-World Staff Assignment Problem. In: *International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 541–554, 2019.

Assigning staff to engagements according to hard constraints while optimizing several objectives is a task encountered by many companies on a regular basis. Simplified versions of such assignment problems are NP-hard. Despite this, a typical approach to solving them consists of formulating them as mixed integer programming (MIP) problems and using a state-of-the-art solver to get solutions that closely approximate the optimum. In this paper, we consider a complex real-world staff assignment problem encountered by the professional service company KPMG, with the goal of finding an algorithm that solves it faster and with a better solution than a commercial MIP solver. We follow the evolutionary algorithm (EA) metaheuristic and design a search heuristic which iteratively improves a solution using domain-specific mutation operators. Furthermore, we use a flow algorithm to optimally solve a subproblem, which tremendously reduces the search space for the EA. For our real-world instance of the assignment problem, given the same total time budget of 100 hours, a parallel EA approach finds a solution that is only 1.7% away from an upper bound for the (unknown) optimum within under five hours, while the MIP solver Gurobi still has a gap of 10.5 %.

[14] Doerr, B., Krejca, M. S., Significance-based Estimation-of-Distribution Algorithms. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 1483–1490, 2018.

Estimation-of-distribution algorithms (EDAs) are randomized search heuristics that maintain a stochastic model of the solution space. This model is updated from iteration to iteration based on the quality of the solutions sampled according to the model. As previous works show, this short-term perspective can lead to erratic updates of the model, in particular, to bit-frequencies approaching a random boundary value. This can lead to significant performance losses. In order to overcome this problem, we propose a new EDA that takes into account a longer history of samples and updates its model only with respect to information which it classifies as statistically significant. We prove that this significance-based compact genetic algorithm (sig-cGA) optimizes the common benchmark functions OneMax and LeadingOnes both in $O(n \log n)$ time, a result shown for no other EDA or evolutionary algorithm so far. For the recently proposed scGA – an EDA that tries to prevent erratic model updates by imposing a bias to the uniformly distributed model – we prove that it optimizes OneMax only in a time exponential in the hypothetical population size $1/\rho$.

[15] Kötzing, T., Krejca, M. S., First-Hitting Times Under Additive Drift. In: *Parallel Problem Solving From Nature (PPSN)*, pp. 92–104, 2018.

For the last ten years, almost every theoretical result concerning the expected run time of a randomized search heuristic used drift theory, making it the arguably most important tool in this domain. Its success is due to its ease of use and its powerful result: drift theory allows the user to derive bounds on the expected first-hitting time of a random process by bounding expected local changes of the process – the drift. This is usually far easier than bounding the expected first-hitting time directly. Due to the widespread use of drift theory, it is of utmost importance to have the best drift theorems possible. We improve the fundamental additive, multiplicative, and variable drift theorems by stating them in a form as general as possible and providing examples of why the restrictions we keep are still necessary. Our additive drift theorem for upper bounds only requires the process to be nonnegative, that is, we remove unnecessary restrictions like a finite, discrete, or bounded search space. As corollaries, the same is true for our upper bounds in the case of variable and multiplicative drift

[16] Kötzing, T., Krejca, M. S., First-Hitting Times for Finite State Spaces. In: *Parallel Problem Solving From Nature (PPSN)*, pp. 79–91, 2018.

One of the most important aspects of a randomized algorithm is bounding its expected run time on various problems. Formally speaking, this means bounding the expected first-hitting time of a random process. The two arguably most popular tools to do so are the fitness level method and drift theory. The fitness level method considers arbitrary transition probabilities but only allows the process to move toward the goal. On the other hand, drift theory allows the process to move into any direction as long as it moves closer to the goal in expectation; however, this tendency has to be monotone and, thus, the transition probabilities cannot be

arbitrary. We provide a result that combines the benefit of these two approaches: our result gives a lower and an upper bound for the expected first-hitting time of a random process over $\{0, \ldots, n\}$ that is allowed to move forward and backward by 1 and can use arbitrary transition probabilities. In case that the transition probabilities are known, our bounds coincide and yield the exact value of the expected first-hitting time. Further, we also state the stationary distribution as well as the mixing time of a special case of our scenario.

[17] Bläsius, T., Eube, J., Feldtkeller, T., Friedrich, T., Krejca, M. S., Lagodzinski, J. A. G., Rothenberger, R., Severin, J., Sommer, F., Trautmann, J., Memory-restricted Routing With Tiled Map Data. In: *Systems, Man, and Cybernetics (SMC)*, pp. 3347–3354, 2018.

Modern routing algorithms reduce query time by depending heavily on preprocessed data. The recently developed Navigation Data Standard (NDS) enforces a separation between algorithms and map data, rendering preprocessing inapplicable. Furthermore, map data is partitioned into tiles with respect to their geographic coordinates. With the limited memory found in portable devices, the number of tiles loaded becomes the major factor for run time. We study routing under these restrictions and present new algorithms as well as empirical evaluations. Our results show that, on average, the most efficient algorithm presented uses more than 20 times fewer tile loads than a normal A*.

[18] Krejca, M. S., Witt, C., Lower Bounds on the Run Time of the Univariate Marginal Distribution Algorithm on OneMax. In: *Foundations of Genetic Algorithms (FOGA)*, pp. 65–79, 2017.

The Univariate Marginal Distribution Algorithm (UMDA), a popular estimation of distribution algorithm, is studied from a run time perspective. On the classical OneMax benchmark function, a lower bound of $\Omega(\mu\sqrt{n} + n \log n)$, where μ is the population size, on its expected run time is proved. This is the first direct lower bound on the run time of the UMDA. It is stronger than the bounds that follow from general black-box complexity theory and is matched by the run time of many evolutionary algorithms. The results are obtained through advanced analyses of the stochastic change of the frequencies of bit values maintained by the algorithm, including carefully designed potential functions. These techniques may prove useful in advancing the field of run time analysis for estimation of distribution algorithms in general.

[19] Arndt, T., Hafner, D., Kellermeier, T., Krogmann, S., Razmjou, A., Krejca, M. S., Rothenberger, R., Friedrich, T., Probabilistic Routing for On-Street Parking Search. In: *European Symposium on Algorithms (ESA)*, pp. 6:1–6:13, 2016.

An estimated 30% of urban traffic is caused by search for parking spots. Traffic could be reduced by suggesting effective routes leading along potential parking spots. In this paper, we formalize parking search as a probabilistic problem on a road graph and show that it is NP-complete. We explore heuristics that optimize for the driving duration and the walking distance to the destination. Routes are constrained to reach a certain probability threshold of finding a spot. Empirically estimated probabilities of successful parking attempts are provided by TomTom on a per-street basis. We release these probabilities as a dataset of about 80,000 roads covering the Berlin area. This allows to evaluate parking search algorithms on a real road network with realistic probabilities for the first time. However, for many other areas, parking probabilities are not openly available. Because they are effortful to collect, we propose an algorithm that relies on conventional road attributes only. Our experiments show that this algorithm comes close to the baseline by a factor of 1.3 in our cost measure. This leads to the conclusion that conventional road attributes may be sufficient to compute reasonably good parking search routes.

[20] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., The Benefit of Recombination in Noisy Evolutionary Search. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 161–162, 2016.

Practical optimization problems frequently include uncertainty about the quality measure, for example due to noisy evaluations. Thus, they do not allow for a straightforward application of traditional optimization techniques. In these settings, randomized search heuristics such as evolutionary algorithms are a popular choice because they are often assumed to exhibit some kind of resistance to noise. Empirical evidence suggests that some algorithms, such as estimation of distribution algorithms (EDAs) are robust against a scaling of the noise intensity, even without resorting to explicit noise-handling techniques such as resampling. In this paper, we want to support such claims with mathematical rigor. We introduce the concept of graceful scaling in which the run time of an algorithm scales polynomially with noise intensity. We study a monotone fitness function over binary strings with additive noise taken from a Gaussian distribution. We show that myopic heuristics cannot efficiently optimize the function under arbitrarily intense noise without any explicit noise-handling. Furthermore, we prove that using a population does not help. Finally we show that a simple EDA called the Compact Genetic Algorithm can overcome the shortsightedness of mutation-only heuristics to scale gracefully with noise. We conjecture that recombinative genetic algorithms also have this property.

[21] Dang, D.-C., Friedrich, T., Krejca, M. S., Kötzing, T., Lehre, P. K., Oliveto, P. S., Sudholt, D., Sutton, A. M., Escaping Local Optima with Diversity Mechanisms and Crossover. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 645–652, 2016.

Population diversity is essential for the effective use of any crossover operator. We compare seven commonly used diversity mechanisms and prove rigorous run time bounds for the $(\mu + 1)$ GA using uniform crossover on the fitness function $Jump_k$. All previous results in this context only hold for unrealistically low crossover probability $p_c = O(k/n)$, while we give analyses for the setting of constant $p_c < 1$ in all but one case. Our bounds show a dependence on the problem size n, the jump length k, the population size μ, and the crossover probability p_c. For the typical case of constant $k > 2$ and constant p_c, we can compare the resulting expected optimisation times for different diversity mechanisms assuming an optimal choice of μ: $O(n^{k-1})$ for duplicate elimination/minimisation, $O(n^2 \log n)$ for maximising the convex hull, $O(n \log n)$ for det. crowding (assuming $p_c = k/n$), $O(n \log n)$ for maximising the Hamming distance, $O(n \log n)$ for fitness sharing, $O(n \log n)$ for the single-receiver island model. This proves a sizeable advantage of all variants of the $(\mu+1)$ GA compared to the (1+1) EA, which requires $\Theta(n^k)$. In a short empirical study we confirm that the asymptotic differences can also be observed experimentally.

[22] Friedrich, T., Kötzing, T., Krejca, M. S., Nallaperuma, S., Neumann, F., Schirneck, M., Fast Building Block Assembly by Majority Vote Crossover. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 661–668, 2016.

Different works have shown how crossover can help with building block assembly. Typically, crossover might get lucky to select good building blocks from each parent, but these lucky choices are usually rare. In this work we consider a crossover operator which works on three parent individuals. In each component, the offspring inherits the value present in the majority of the parents; thus, we call this crossover operator majority vote. We show that, if good components are sufficiently prevalent in the individuals, majority vote creates an optimal individual with high probability. Furthermore, we show that this process can be amplified: as long as components are good independently and with probability at least $1/2 + \delta$, we require only $O(\log 1/\delta + \log \log n)$ successive stages of majority vote to create an optimal individual with high probability! We show how this applies in two scenarios. The first scenario is the Jump test function. With sufficient diversity, we get an optimization time of $O(n \log n)$ even for jump sizes as large as $O(n^{(1/2-\epsilon)})$. Our second scenario is a family of vertex cover instances. Majority vote optimizes this family efficiently, while local searches fail and only highly specialized two-parent crossovers are successful.

[23] Friedrich, T., Kötzing, T., Krejca, M. S., EDAs cannot be Balanced and Stable. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 1139–1146, 2016.

Estimation of Distribution Algorithms (EDAs) work by iteratively updating a distribution over the search space with the help of samples from each iteration. Up to now, theoretical analyses of EDAs are scarce and present run time results for specific EDAs. We propose a new framework for EDAs that captures the idea of several known optimizers, including PBIL, UMDA, λ-MMASIB, cGA, and $(1, \lambda)$-EA. Our focus is on analyzing two core features of EDAs: a balanced EDA is sensitive to signals in the fitness; a stable EDA remains uncommitted under a biasless fitness function. We prove that no EDA can be both balanced and stable. The LeadingOnes function is a prime example where, at the beginning of the optimization, the fitness function shows no bias for many bits. Since many well-known EDAs are balanced and thus not stable, they are not well-suited to optimize LeadingOnes. We give a stable EDA which optimizes LeadingOnes within a time of $O(n \log n)$.

[24] Dang, D.-C., Lehre, P. K., Friedrich, T., Kötzing, T., Krejca, M. S., Oliveto, P. S., Sudholt, D., Sutton, A. M., Emergence of Diversity and its Benefits for Crossover in Genetic Algorithms. In: *Parallel Problem Solving From Nature (PPSN)*, pp. 890–900, 2016.

Population diversity is essential for avoiding premature convergence in Genetic Algorithms (GAs) and for the effective use of crossover. Yet the dynamics of how diversity emerges in populations are not well understood. We use rigorous runtime analysis to gain insight into population dynamics and GA performance for a standard $(\mu + 1)$ GA and the $Jump_k$ test function. By studying the stochastic process underlying the size of the largest collection of identical genotypes we show that the interplay of crossover followed by mutation may serve as a catalyst leading to a sudden burst of diversity. This leads to improvements of the expected optimisation time of order $\Omega(n/\log n)$ compared to mutation-only algorithms like the $(1 + 1)$ EA.

[25] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., Graceful Scaling on Uniform versus Steep-Tailed Noise. In: *Parallel Problem Solving From Nature (PPSN)*, pp. 761–770, 2016.

Recently, different evolutionary algorithms (EAs) have been analyzed in noisy environments. The most frequently used noise model for this was additive posterior noise (noise added after the fitness evaluation) taken from a Gaussian distribution. In particular, for this setting it was shown that the $(\mu + 1)$-EA on OneMax does not scale gracefully (higher noise cannot efficiently be compensated by higher μ). In this paper we want to understand whether there is anything special about the Gaussian distribution which makes the $(\mu + 1)$-EA not scale gracefully. We keep the setting of posterior noise, but we look at other distributions. We see that for exponential tails the $(\mu + 1)$-EA on OneMax does also not scale gracefully, for similar reasons as in the case of Gaussian noise. On the other hand, for uniform distributions (as well as other, similar distributions) we see that the $(\mu + 1)$-EA on OneMax does scale gracefully, indicating the importance of the noise model.

[26] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., Robustness of Ant Colony Optimization to Noise. In: *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 17–24, 2015. Best-Paper Award (ACO/SI Track).

Recently Ant Colony Optimization (ACO) algorithms have been proven to be efficient in uncertain environments, such as noisy or dynamically changing fitness functions. Most of these analyses focus on combinatorial problems, such as path finding. We analyze an ACO algorithm in a setting where we try to optimize the simple OneMax test function, but with additive posterior noise sampled from a Gaussian distribution. Without noise the classical $(\mu + 1)$-EA outperforms any ACO algorithm, with smaller μ being better; however, with large noise, the $(\mu + 1)$-EA fails, even for high values of μ (which are known to help against small noise). In this paper we show that ACO is able to deal with arbitrarily large noise in a graceful manner, that is, as long as the evaporation factor μ is small enough dependent on the parameter δ^2 of the noise and the dimension n of the search space $(p = o(1/(n(n + \delta \log n)^2 \log n)))$, optimization will be successful.

[27] Friedrich, T., Kötzing, T., Krejca, M. S., Sutton, A. M., The Benefit of Recombination in Noisy Evolutionary Search. In: *International Symposium of Algorithms and Computation (ISAAC)*, pp. 140–150, 2015.

Practical optimization problems frequently include uncertainty about the quality measure, for example due to noisy evaluations. Thus, they do not allow for a straightforward application of traditional optimization techniques. In these settings meta-heuristics are a popular choice for deriving good optimization algorithms, most notably evolutionary algorithms which mimic evolution in nature. Empirical evidence suggests that genetic recombination is useful in uncertain environments because it can stabilize a noisy fitness signal. With this paper we want to support this claim with mathematical rigor. The setting we consider is that of noisy optimization. We study a simple noisy fitness function that is derived by adding Gaussian noise to a monotone function. First, we show that a classical evolutionary algorithm that does not employ sexual recombination (the $(\mu + 1)$-EA) cannot handle the noise efficiently, regardless of the population size. Then we show that an evolutionary algorithm which does employ sexual recombination (the Compact Genetic Algorithm, short: cGA) can handle the noise using a graceful scaling of the population.